VERSION WITH MARKINGS TO SHOW CHANGES MADE

In the Specification

The paragraph beginning at page 2, line 3, has been amended as follows:

Referring now to FIG 1, one common arrangement of providing console device access to server appliances in a network 100 is depicted. As depicted, a plurality of server appliances 102 are connected to each other via a network medium such as an Ethernet connection. In addition, each server appliance 102 includes a connection for a keyboard, mouse, and video terminal (or other suitable display unit). Although FIG 1 indicates physically distinct connections for the keyboard, mouse, and video, these connections may be provided using a single cable or wire[.]. The KVM (keyboard-video-mouse) connections from each server appliance 102 are provided to a switch 104, which is itself connected to a keyboard 106, a mouse 108, and a video terminal 110. Using the arrangement, the network administrator can transmit and receive console interactions to/from each of the server appliances 102 by appropriate setting of switch 104.

The paragraph beginning at page 2, line 13, has been amended as follows:

An important goal in the design of any server appliance is to minimize the cost of the system and to make it easy to set up and operate. Additionally, for systems such as the systems described herein, which are intended to be aggregated in large numbers, it is highly desirable to minimize the overall cost of the cluster by reducing or eliminating the amount of supporting hardware required.

The paragraph beginning at page 2, line 18, has been amended as follows:

One of the sources of cost and complexity in current server appliances is the extra cabling and switching required to provide console access to the server. This extra cabling can become a major issue in server farm environments where large numbers of server appliances are congregated together in a single rack or set of racks. Therefore, it would be highly desirable to provide a system and method for transmitting console traffic to selected server appliances without requiring additional cabling and other hardware. It would be further desirable if the implemented solution leveraged, to the extent possible, existing features of the network to minimize the cost and complexity of implementing the invention.

The paragraph beginning at page 2, line 29, has been amended as follows:

The problems identified above are in large part addressed by a system and method in which console interactions are communicated to and from network server appliances over the network to which the servers are connected. In one embodiment, the system includes a server appliance having software, firmware, hardware, or a combination thereof, that is configured to re-direct serial port interactions to a network port. The server appliance includes a mechanism for transmitting and receiving console data and control information via the network. The system further includes means for accepting and displaying console traffic that is sent over the network by a server appliance and transmitting commands entered by a user back to the server appliance for processing.

The paragraph beginning at page 3, line 7, has been amended as follows:

In one embodiment, the firmware of the server appliance is enabled to capture console interactions directed to one of the server appliance's serial ports and re-direct these interactions over the network. The firmware of the server appliance may use existing firmware features, such as the network features of DHCP support, to transmit and receive the console traffic via the network. Another system attached to the network is configured to receive the console interactions from the network and display them on a directly attached console device. This system may include an application that uses network interactions similar to Telnet to receive console interactions from the network, and serial line communications features similar to a serial terminal emulator such as the Minicom emulator [~~from IBM~~]. In one embodiment, this application is configured to display multiple windows, where each window displays the console communications of a corresponding server appliance.

The paragraph beginning at page 7, line 4, has been amended as follows:

As depicted in FIG 3, system 300 further includes a console server 320 that provides console support for each server appliance 200 connected to the network. Console server 320 typically includes its own network interface card 321 through which IP formatted packets with the appropriate destination IP address (the IP address of console server 320) are detected. Console server 320 may further include an application program identified as emulator 324 that is

responsible for enabling communication with the remote[~~r~~] server appliances and for interpreting the serial data in each packet received from a server appliance.

The paragraph beginning at page 7, line 11, has been amended as follows:

Emulator 324 typically has a serial portion that includes the serial communication features of a standard serial communication emulator (such as the *minicom* emulator [~~from IBM Corporation~~]) that enables a device to communicate with a host over a serial line. These serial communication features of emulator 324 understand the serial protocol and control sequences that the server appliance uses to communicate with a console device attached to its serial port. In addition, emulator 324 has a network portion that includes network communication features enabling console server 320 to communicate with a host (i.e., server appliance) device using a network protocol such as IP. In one embodiment, the network communication features of emulator 324 are taken from the Telnet specification, which is designed to enable a remote device to communicate with a host over IP. The Telnet specification is documented in *Telnet Protocol Specification*, Internet RFC854 (Network Working Group 1983), available online at http://www.faqs.org/rfcs/rfc854.html and incorporated by reference herein. In a typical Telnet session, the remote device becomes a dumb terminal for the host device to which it is connected. Since Telnet has been in existence in various forms since as early as 1971 and is in wide use, it provides a familiar vehicle for enabling the network communication. Thus, emulator 324 may pattern its IP communication mechanisms after the communications routines of Telnet.

The paragraph beginning at page 7, line 27, has been amended as follows:

Console server 320 further includes an operating system 322 that provides an interface between emulator 324 and a keyboard 326, a video terminal 328, and a mouse 330. Emulator 324 uses the interface provided by OS 322 to direct the console interactions received from the network to video terminal [~~324~~] 328. Emulator 324 uses the interface provided by OS 322 to accept input from keyboard [~~324~~] 326 and mouse [~~300~~] 330 and to transmit this input to the server appliance over the network using NIC 321.

In the Claims

Claim 16 has been amended as follows:

16. (Amended) The system of claim 15, wherein the emulator is configured to display a first window for displaying console transactions occurring between the first server appliance and the console server and a second window for displaying console transactions occurring between the second server appliance and the console server.

In the Abstract

The Abstract has been amended as follows:

A data processing network in which console interactions are communicated to and from server appliances over the network. The system may include a server appliance configured to re-direct serial port transactions to a network port. The service appliance may include a mechanism for transmitting and receiving console data and control information via the network. The system further includes a console server for accepting and displaying console traffic that is sent over the network by a server appliance and for transmitting commands entered by a user back to the server appliance for processing.